



Universitatea Politehnica București  
Facultatea de Automatică și Calculatoare  
Departamentul de Automatică și Ingineria Sistemelor

# Sisteme integrate de conducere

## Proiect

Sistem de monitorizare al temperaturii și  
control al intensității luminii

Student: Molet Alexandru-Valentin  
Grupa: 341 B2  
Îndrumător: Prof. Ciprian Lupu

## Cuprins

1. Introducere .....	2
2. Lista componentelor.....	2
3. Monitorizarea temperaturii .....	3
4. Controlul intensității luminii.....	6
5. Comutarea dintre monitorizare și control .....	9
6. Rezultatele implementării .....	10
7. Siguranța și protecția sistemului .....	13
8. Concluzii .....	14
Bibliografie.....	15

## 1. Introducere

În această lucrare, s-a proiectat un sistem format din două funcționalități: monitorizarea temperaturii dintr-o încăpăre și controlul nivelului de lumină din aceasta.

Achiziția temperaturii se realizează prin intermediul unui senzor de temperatură digital, DHT11. Pentru măsurarea nivelului de lumină, se utilizează o fotorezistență de tip 5528. Afișarea celor două mărimi măsurate se face în cadrul unui ecran LCD 1602 cu Interfața I2C și Backlight Albastru.

Controlul nivelului de lumină se implementează printr-un algoritm de reglare de tip PI: proporțional integrativ. Sursa de lumină reglată este reprezentată de un LED de culoare galbenă, care își schimbă intensitatea luminii în funcție de referința dată de un potențiomtru de 10 k $\Omega$ . Tot prin intermediul ecranului LCD, se pot vizualiza stările buclei de reglare: referința, comanda și ieșirea.

Atât monitorizarea cât și reglarea se realizează printr-o aplicație software scrisă în limbajul Arduino. Aceasta funcționează în timp real și este încărcată într-un microprocesor Arduino Nano.

## 2. Lista componentelor

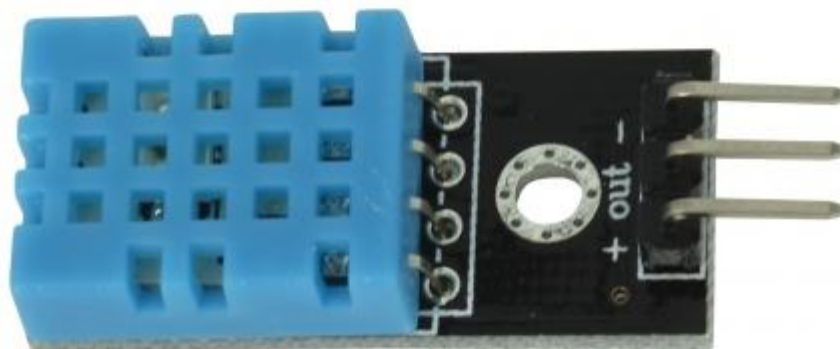
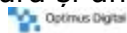
Lista completă cu toate elementele folosite pentru realizarea acestui proiect se găsește mai jos, în tabelul 2.1:

Nr. crt.	Denumirea componentei	Cantitate
1	Rezistor 220 $\Omega$	4
2	Rezistor 10 k $\Omega$	2
3	Fotorezistor tip 5528	1
4	LCD 1602 cu Interfața I2C și Backlight Albastru	1
5	Modul Senzor de Temperatură și Umiditate DHT11	1
6	Arduino Nano	1
7	Cablu USB	1
8	Breadboard din plastic cu 830 contacte nichelate	1
9	Buton Pushbutton	1
10	Potentiometru 10 k $\Omega$	1
11	LED roșu	1
12	LED verde	1
13	LED galben	1
14	LED alb	1
15	Cabluri tată-tată	19
16	Cabluri mamă-tată	4
17	Cabluri mamă-mamă	3

*Tabelul 2.1. Lista componentelor utilizate pentru construirea sistemului*

### 3. Monitorizarea temperaturii

Pentru monitorizarea temperaturii dintr-o cameră se folosește un senzor de temperatură și umiditate DHT11, ca în figura 3.1:



*Figura 3.1. Senzorul de temperatură și umiditate DHT11*

S-a ales acest dispozitiv de măsurare, deoarece este un senzor de înaltă performanță, asigurând o bună fiabilitate și stabilitate.

Caracteristici tehnice:

- Tensiune de alimentare: 3.3V - 5V;
- Curent: 2.5mA, maxim;
- Gama de măsurare a umidității: 20% - 95% RH;
- Acuratețea măsurării umidității:  $\pm 5\%$  RH;
- Gama de măsurare a temperaturii: 0 °C - 60 °C;
- Acuratețea măsurării temperaturii:  $\pm 2$  °C;
- Nu funcționează sub 0 °C;
- Dimensiuni: 32mm x 14mm.

Temperatura este măsurată de un termistor NTC, iar umiditatea relativă este măsurată folosind un senzor capacitiv. Aceste elemente sunt pre-calibrate, iar ieșirea este oferită ca semnal digital. Acest senzor este compatibil cu cele mai populare plăcuțe de dezvoltare, precum Arduino. DHT11 este convenabil, oferind precizie bună, simplitate în utilizare și dimensiuni reduse la un preț mic. Senzorul are inclus un rezistor de pull-up, astfel încât nu mai este nevoie de componente adiționale. [\[\\*\\*\\*-21\]](#)

Pentru realizarea conexiunilor hardware necesare, se urmărește schema de conectare de mai jos:

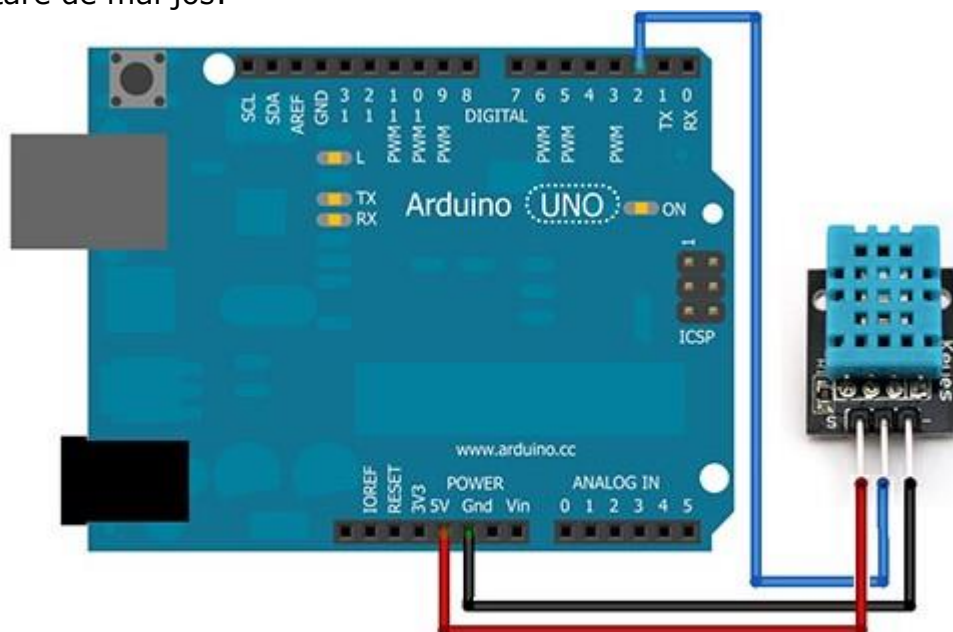


Figura 3.2. Schema de conectare dintre Arduino și DHT11

Din figura 3.2 se observă simplitatea conectării senzorului cu placa de dezvoltare Arduino, circuitul necesitând doar trei fire. Pin-ul marcat cu "+" se conectează la pin-ul de 5 V al plăcii. Pin-ul marcat cu "OUT" se conectează la pin-ul digital 2, unde vor intra în microprocesor datele mărimii măsurate. Pin-ul marcat cu "-" se conectează la pinul GND.

Pentru a utiliza acest senzor în aplicația software, Arduino IDE are nevoie de o bibliotecă special creată pentru acesta. În cadrul acestui proiect, s-a folosit biblioteca „SimpleDHT.h”, care permite citirea temperaturii de la DHT11. Pentru a citi fără erori, se alege 115200 bps pentru comunicarea serială. Totodată, în interiorul structurii repetitive, la final, se așteaptă 1500 de milisecunde, după fiecare iterație, deoarece timpul de eșantionare al senzorului este de un 1 Hz.

Pentru a afișa mărimea măsurată, se utilizează un modul LCD, ca în figura 3.3:



Figura 3.3. Modul LCD ce afișează 2 rânduri a câte 16 caractere

Acest modul comunică pe interfața I<sup>2</sup>C sau circuit inter-integrat, cum se poate observa în figura 3.4:



Figura 3.4. Interfața I<sup>2</sup>C

Este un tip de magistrală pentru transmisia de date serială master-slave, utilizată între circuitele integrate digitale, cum ar fi: microcontrolere, memorii sau convertoare. A fost dezvoltată inițial de către firma olandeză Philips, în anul 1982. Magistrala este compusă din două linii: linia de ceas serială, SCL, și linia de date serială, SDA. [\*\*\*-21]

Caracteristicile tehnice ale modului sunt:

- Tensiune de alimentare: 5 V;
- Curent: 1.1 mA;
- Tensiune de alimentare backlight: 4.2 V;
- Curent backlight: 100 mA;
- Dimensiuni: 80 mm x 36 mm x 12 mm.

Interfața I<sup>2</sup>C oferă un avantaj: masterul se poate conecta la mai multe slave-uri. De aici rezultă că modulul are nevoie doar de două conexiuni, SCL și SDA, cu placa de dezvoltare, pentru a transfera datele, plus conexiunile necesare pentru masă și alimentare. [\*\*\*-21]

Conectarea la placa de dezvoltare se face astfel:

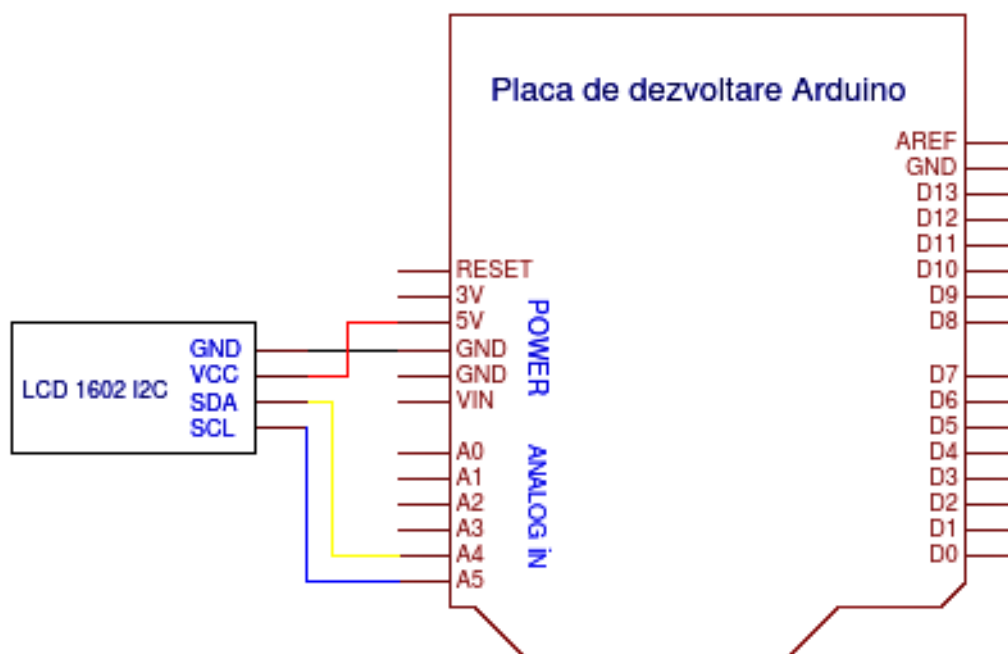


Figura 3.5. Conectarea modului LCD la placa de dezvoltare

În figura 3.5 se observă că pinii SDA și SCL corespund pinilor analogici A4, respectiv A5. Alimentarea modului se face cu o tensiune de 5 V curent continuu. Pentru a programa afișarea datelor pe acest LCD se folosesc două biblioteci necesare: „Wire.h” și „LiquidCrystal\_PCF8574.h”. Adresa I<sup>2</sup>C a LCD-ului este 0x27.

Pe lângă afișarea temperaturii pe ecranul LCD, se mai utilizează 3 LED de culori diferite: roșu, verde și alb. Acestea sunt conectate la pinii digitali 6, 7, respectiv 8. Rolul lor este de a semnaliza disconfortul termic din cameră. Dacă temperatura este mai mare decât 26 °C, atunci este prea cald și se aprinde becul roșu. Dacă valoarea acesteia se află între 21 și 25 °C, atunci temperatura este plăcută în încăperea și se aprinde becul verde. Altfel, se aprinde becul alb, semnalând că este frig în cameră. Pentru a evita arderea diodelor electroluminescente, se leagă câte o rezistență 220  $\Omega$  la anodul fiecăreia pentru a limita voltajul.

#### 4. Controlul intensității luminii

Pentru controlul sistemului, se consideră următoarea structură de reglare automată:

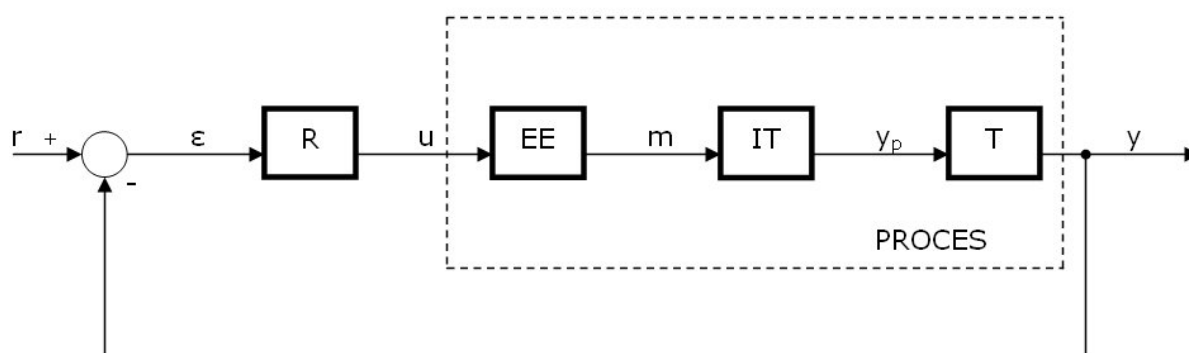


Figure 4.1. SRA standard

În figura 4.1 se observă elementele de bază ale unei bucle de reglare standard. Aceasta are un singur grad de libertate. De asemenea, procesul este un sistem de ordinul I:

$$H(s) = \frac{K}{Ts + 1} \quad (4.1)$$

unde  $K$  și  $T > 0$  sunt factorul de amplificare, respectiv constanta de timp a sistemului. [\[DUM-10\]](#)

Așadar, se utilizează un regulator PI, de forma:

$$H_{PI}(s) = K_r \left( \frac{T_i s + 1}{T_i s} \right) \quad (4.2)$$

Componenta proporțională are următoarea ecuație:

$$u(t) = K_r \varepsilon(t) \quad (4.3)$$

Aceasta, folosită separat, nu va oferi eroare staționară zero și este insensibilă la perturbații. Odată cu creșterea câștigului  $K_r$  crește amplificarea semnalului și suprareglajul, iar eroarea staționară este mai mică. Pentru o valoare prea mare sistemul intră în instabilitate. [\[POP-11\]](#)

Componenta integrativă este reprezentată astfel:

$$u(t) = \frac{K_r}{T_i} \int_0^t \varepsilon(\tau) d\tau \quad (4.4)$$

Componenta integrativă este proporțională atât cu amplitudinea erorii cât și cu durata ei. Integrala reprezintă însumarea tuturor valorilor erorii instantanee de-a lungul timpului. Componenta integrativă va accelera evoluția procesului către valoarea referinței și va înlătura eroarea staționară care apare în urma regulatorului proporțional. Deci integrala va răspunde în funcție de valorile acumulate ale erorii în trecut și poate avea ca efect la momentul prezent apariția suprareglajelor. [\[POP-11\]](#)

Pentru implementarea regulatorului în microcontroller, acesta se discretizează.

Funcția de transfer a regulatorului devine:

$$H_R(z^{-1}) = \frac{q_0 + q_1 z^{-1}}{1 - z^{-1}} \quad (4.5)$$

iar comanda regulatorului va avea forma incrementală:

$$u[k] = u[k-1] + q_0 \varepsilon[k] + q_1 \varepsilon[k-1] \quad (4.6)$$

unde:



$$q_0 = K_r \left( 1 + \frac{T_e}{T_i} \right) \quad (4.7)$$

$$q_1 = -K_r \quad (4.8)$$

$T_e$  reprezintă perioada de eșantionare. Aceste formule se obțin folosind transformarea Tustin:

$$s = \frac{2(1 - z^{-1})}{T_e(1 + z^{-1})} \quad (4.9)$$

Regulatorul de tip PI are un singur grad de libertate. Prin urmare acesta oferă soluția reglării unui singur obiectiv, fiind urmărirea referinței [\[POP-11\]](#). Aceasta este dată de un potențiomtru de 10 kΩ, conectat la intrarea analogică A1. Măsura procesului se realizează prin fotorezistență, conectată la intrarea analogică A0, așa cum sugerează figura 4.2. Comanda este ilustrată printr-un LED de culoare galbenă, conectat la microcontroller prin ieșirea digitală D3.

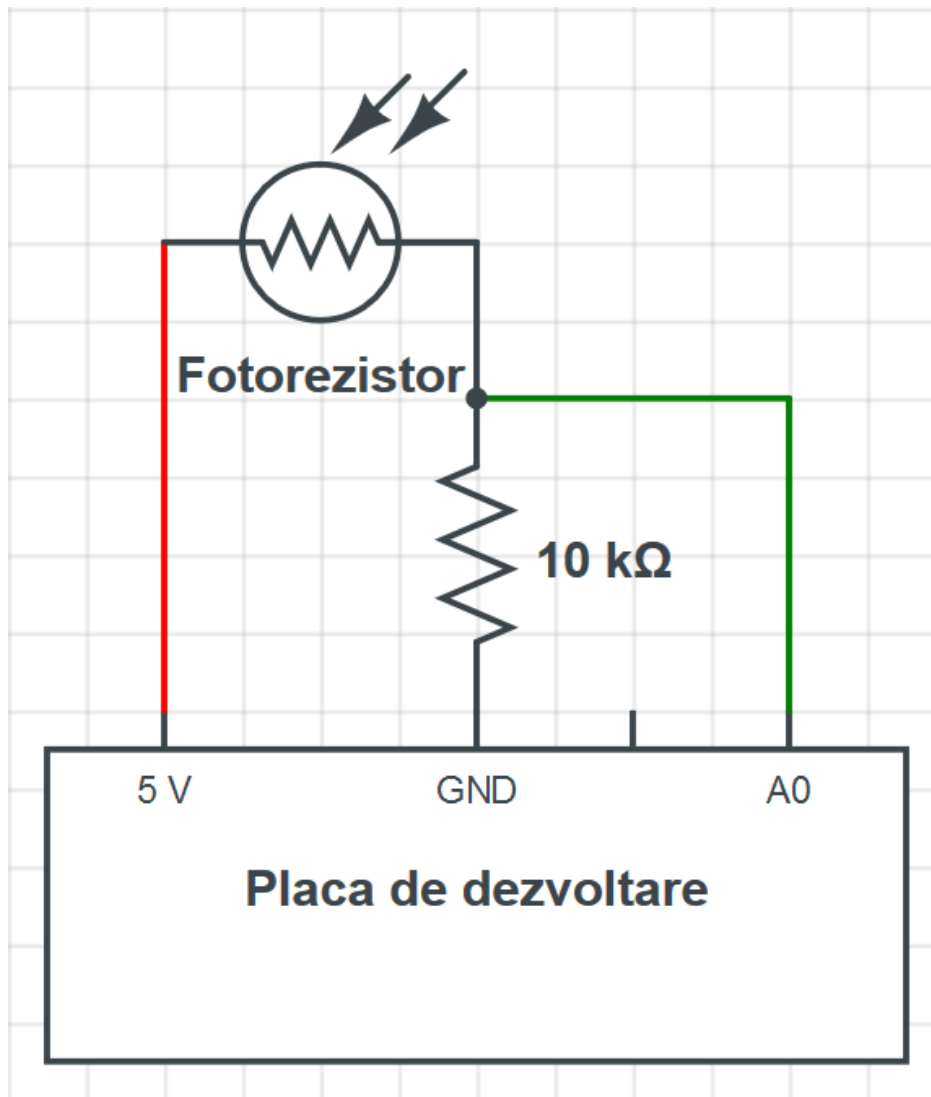


Figura 1.2. Schema de conectare a fotorezistenței

Referința  $r_k$  se află ținându-se cont de intrarea analogică, care are un convertor pe 10 biți. Așadar,  $r_k$  se calculează astfel:

$$r_k = \frac{\text{Valoare senzor}}{1023} \cdot 100 \quad (4.10)$$

datele fiind scalate în procente, între 0% și 100%. Analog și pentru măsura procesului,  $y_k$ . Pentru aplicarea comenzii, se ține cont de convertorul pe 8 biți al ieșirii digitale. Prin urmare, comanda se scrie sub forma:

$$\frac{u_k}{100} \cdot 255 \quad (4.11)$$

Sincronizarea cu perioada de eșantionare,  $T_e$ , se realizează prin intermediul funcției *millis()*, care rulează în fundal și funcționează în permanență. Se utilizează această metodă, pentru că operațiile utile se execută mai rapid decât efectul generat de întârzierea funcției *delay()*. Cele trei stări: referința,  $r_k$ , comanda,  $u_k$  și mărimea procesului,  $y_k$ , sunt afișate pe modulul LCD.

Pentru regulator, s-au ales următoarele constante:

- $K_r = 0,01$ ;
- perioada de eșantionare  $T_e = 0,01$ ;
- $T_i = 0,001$ .

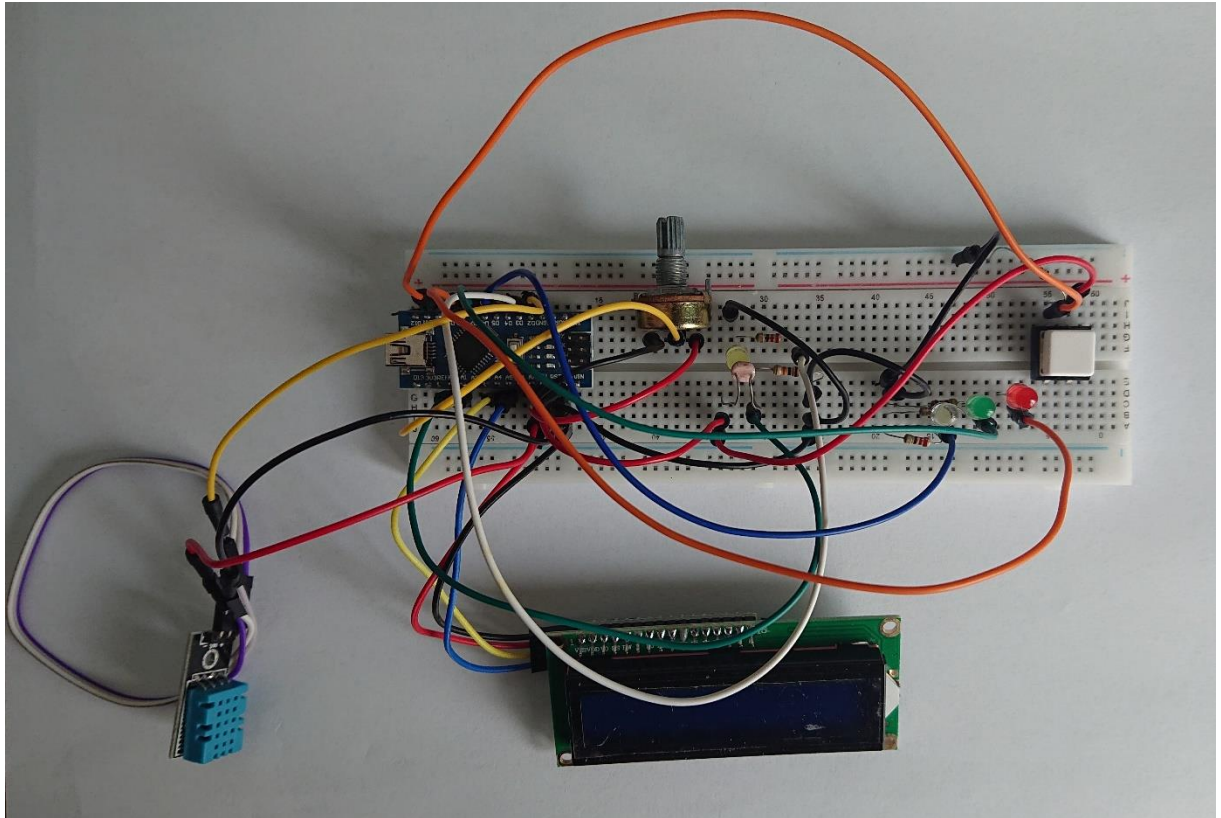
Pentru comunicarea serială, se alege valoarea de 9600 bps.

## 5. Comutarea dintre monitorizare și control

Conectând ambii senzori la același microcontroller, se observă că aceștia nu pot funcționa corect cu aceeași comunicare serială. De aceea, pentru a separa cele două roluri, se folosește un buton auxiliar care comută între control și monitorizare. Se alege, ca mod implicit, reglarea nivelului de lumină. Dacă butonul este apăsat de două ori, se trece la măsurarea temperaturii. Dacă se mai apasă o dată, se revine la controlul luminii. Pentru ca butonul să funcționeze corect, se utilizează un circuit de tip „external pull down resistor” [\*\*\*-21], folosind o rezistență de 10 kΩ. Butonul este conectat la placa de dezvoltare prin intermediul pinului digital D11 și este alimentat la 5 V, curent continuu. Prin această metodă, monitorizarea se face printr-o comunicare serială de 115200 bps, iar controlul cu o comunicare serială de 9600 bps.

## 6. Rezultatele implementării

Circuitul fizic al sistemului este ilustrat în figura 6.1:



*Figura 6.1. Circuitul fizic al sistemului*

Acesta este alimentat la o tensiune de 5 V, curent continuu, prin intermediul plăcii de dezvoltare conectată la sursa de alimentare printr-un cablu USB.

După ce a fost legat la sursa de curent, sistemul din figura 6.1 pornește, inițializând modulul LCD și execută controlul nivelului de lumină al LED-ului galben.

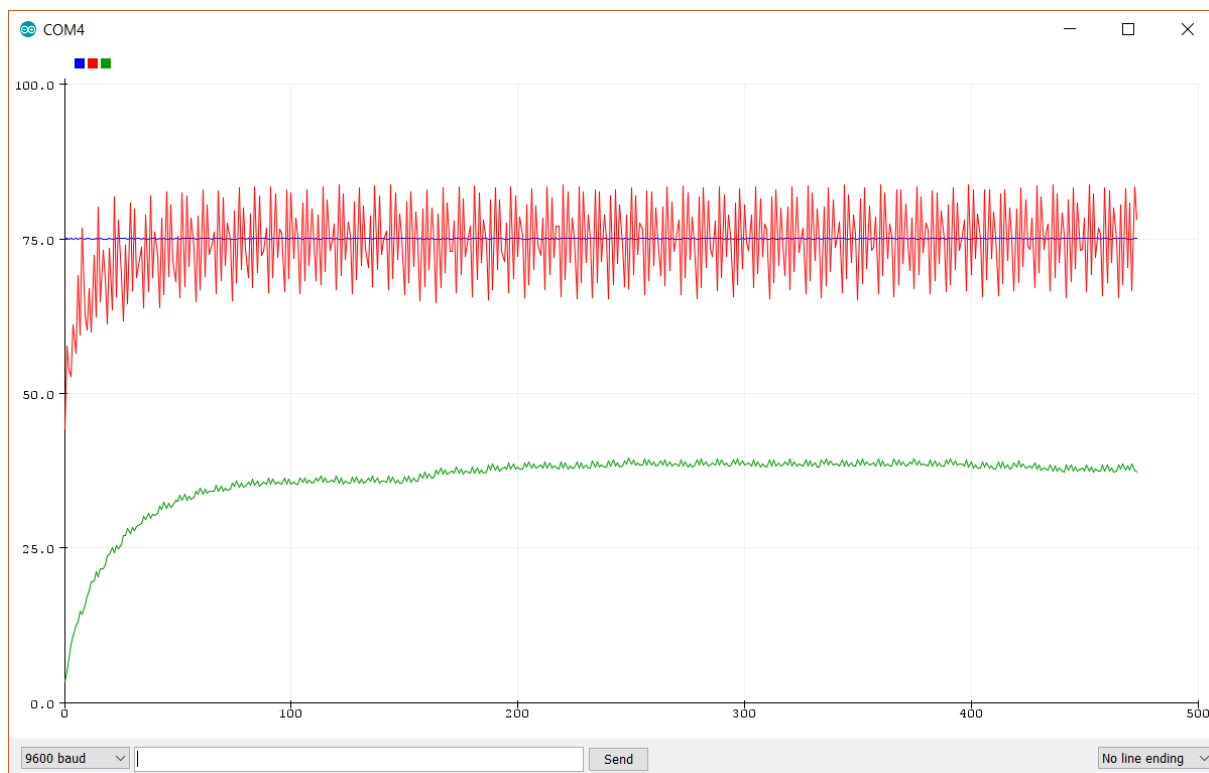


Figura 6.2. Graficul reglării nivelului de lumină al LED-ului

În figura 6.2 se observă evoluția în timp a sistemului. Graficul albastru reprezintă referința dată de potențiometru,  $r_k$ . Cel roșu este mărimea măsurată a procesului,  $y_k$ , iar cel verde este comanda dată LED-ului galben,  $u_k$ . Graficul roșu ilustrează evoluția în timp a sistemului, specifică unei funcții de transfer de ordinul I. De asemenea, demonstrează aplicarea corectă a algoritmului de reglare PI,  $y_k$  urmărind cu succes referința  $r_k$ . Cu toate acestea, se observă prezența perturbației la nivelul măsurării luminii, fotorezistența captând și alte surse de lumină. Această problemă poate fi filtrată prin intermediul unui senzor mai performant.



Figura 6.3. Stările de reglare afișate pe modulul LCD

Sistem de monitorizare al temperaturii și control al intensității luminii

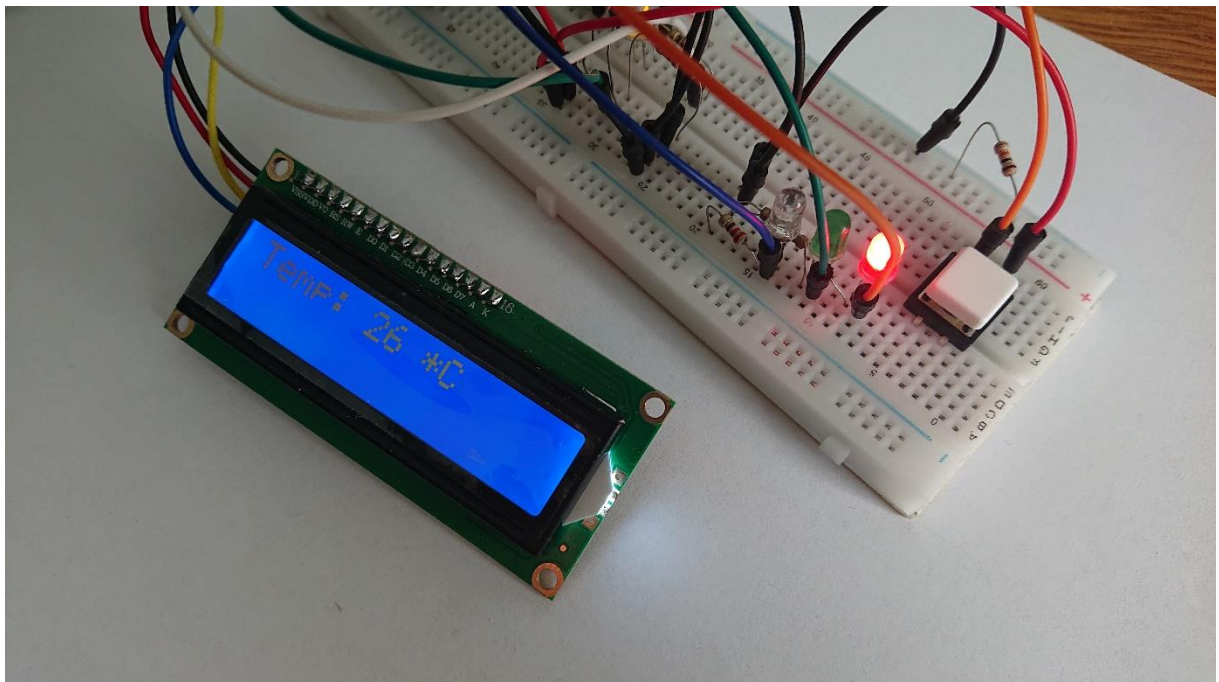
De asemenea, cele trei stări de reglare sunt afișate și pe ecranul LCD, așa cum ilustrează figura 6.3.

Trecând în modul de monitorizare al temperaturii, cu ajutorul butonului, stările de reglare dispar de pe ecran și se afișează temperatura curentă din încăperea, ca în figura 6.4.



*Figura 6.4. Afișarea temperaturii din cameră*

Totodată, se afișează cu ajutorul celorlalte trei LED-uri, impactul temperaturii asupra corpului uman, așa cum ilustrează figurile 6.5 și 6.6.



*Figura 6.5. Semnalarea temperaturii ridicate*



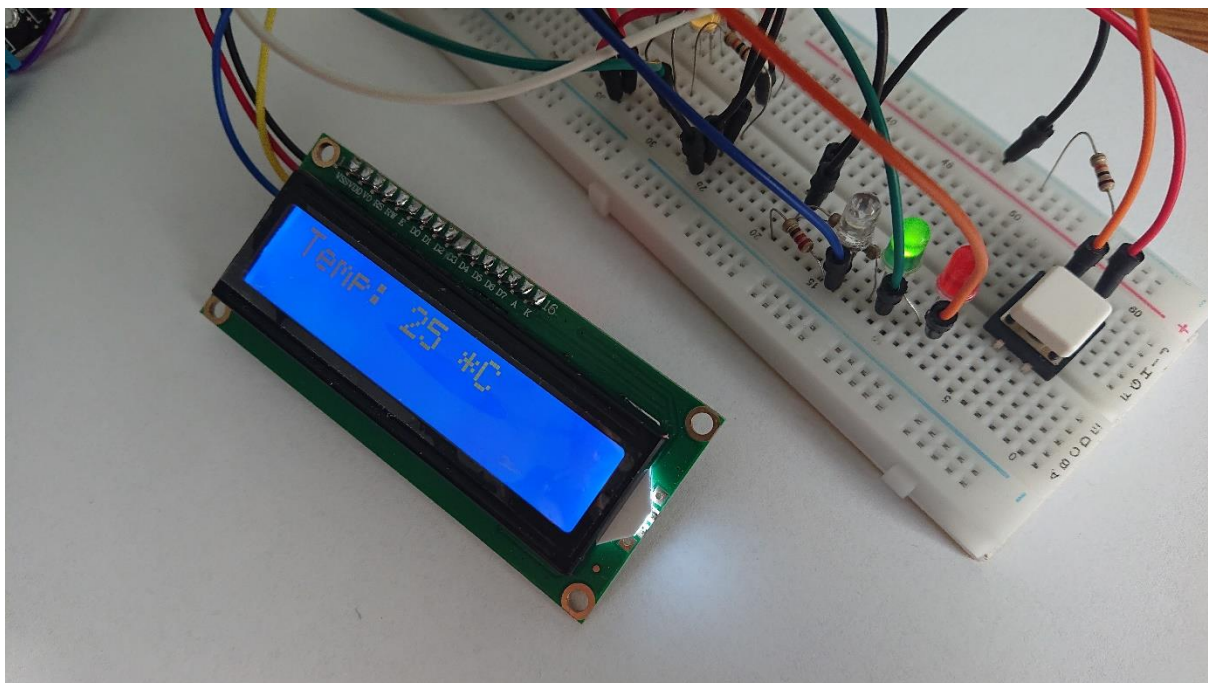


Figura 6.6. Semnalarea temperaturii potrivite

## 7. Siguranța și protecția sistemului

Din punct de vedere al siguranței utilizatorului, acesta este ferit de pericolul de electrocutare, deoarece circuitul funcționează la o tensiune de 5 V și la o intensitate mică. În schimb, dacă sistemul s-ar folosi la nivelul unei camere de apartament sau casă, unde s-ar controla sursa de lumină a încăperii, utilizatorul se poate electrocuta din cauza tensiunii de 220V AC, dacă nu s-a realizat corect împământarea.

Analizând protecția circuitului, acesta se poate acoperi cu o carcasă transparentă pentru a evita deconectarea unui fir. Desigur, carcasa trebuie proiectată astfel încât să se poată alimenta placa de dezvoltare prin borna USB și să se acționeze potențiometrul sau butonul de schimbare a celor două moduri. Pentru protejarea diodelor electroluminescente, s-a utilizat rezistența de 220  $\Omega$ . Dacă s-ar folosi rezistori cu rezistență mai mare, atunci LED-urile nu ar mai funcționa la capacitatea lor maximă și dacă aceștia ar fi mai mici de 220  $\Omega$ , atunci există riscul ca diodele să se ardă din cauza tensiunii ridicate. La nivelul alimentării plăcii de dezvoltare, dacă aceasta se conectează la o sursă de tensiune mai mare de 5 V, există riscul ca procesorul ei să se ardă. De asemenea, celelalte elemente din circuit sunt legate de dispozitivul Arduino. Dacă alimentarea acestuia este întreruptă, atunci întregul circuit nu va mai funcționa. Totodată, dacă se aplică o forță prea mare la apăsarea butonului, atunci acesta se strică și nu se mai poate schimba între modul de control al luminii și cel al monitorizării temperaturii.

În figura 6.2, se observă cum măsurarea mărimii procesului,  $y_k$ , nu are cea mai bună precizie. Pentru a reduce aceste oscilații, se poate folosi un fotorezistor mai avansat sau se poate acoperi fotorezistorul curent și LED-ul galben cu o carcasă opacă, suficient de mare pentru a cuprinde cele două componente, astfel încât să se reducă lumina provenită din exterior pentru a reduce perturbațiile.

În ultimul rând, la nivelul programului software, s-a limitat comanda  $u_k$  în intervalul  $[0, 100]$  % pentru a evita pericolul de defectare a LED-ului galben, în cazul în care valoarea comenzii devine prea mare.

## **8. Concluzii**

În cadrul acestui proiect, s-a urmărit proiectarea unei aplicații software, precum și implementarea hardware pentru monitorizarea și controlul unei instalații de mici dimensiuni.

Din punct de vedere al monitorizării, s-au implementat structurile de achiziție a celor două mărimi măsurate: temperatura și intensitatea luminii. Totodată, s-au filtrat și normat datele achiziționate în funcție de convertoarele analogice pe 10 biți și cele digitale pe 8 biți. De asemenea, s-au afișat mărimile măsurate cu ajutorul unui modul LCD cu interfața I<sup>2</sup>C.

Pentru control, s-a identificat funcția de transfer a procesului și s-a proiectat algoritmul de reglare PI pentru urmărirea referinței, alegând corect parametrii regulatorului și perioada de eșantionare. S-a implementat bucla de reglare a nivelului de lumină, atât numeric, prin intermediul discretizării, cât și fizic. De asemenea, se afișează pe ecranul LCD stările buclei de reglare: referința, ieșirea și comanda.

În final, în cadrul implementării aplicației software, s-a luat în calcul funcționarea programului în timp real.

## Bibliografie

- [\*\*\*-21] [https://www.optimusdigital.ro/ro/senzori-senzori-de-temperatura/99-senzor-de-temperatura-si-si-umiditate-dht11.html?search\\_query=dht11&results=19](https://www.optimusdigital.ro/ro/senzori-senzori-de-temperatura/99-senzor-de-temperatura-si-si-umiditate-dht11.html?search_query=dht11&results=19)  
(Consultat la data: 04-05-2021)
- [\*\*\*-21] [https://www.optimusdigital.ro/ro/optoelectronice-lcd-uri/2894-lcd-cu-interfata-i2c-si-backlight-albastru.html?search\\_query=i2c&results=386](https://www.optimusdigital.ro/ro/optoelectronice-lcd-uri/2894-lcd-cu-interfata-i2c-si-backlight-albastru.html?search_query=i2c&results=386)  
(Consultat la data: 04-05-2021)
- [\*\*\*-21] <https://create.arduino.cc/projecthub/muhammad-aqib/arduino-button-tutorial-using-arduino-digitalread-function-08adb5>  
(Consultat la data: 04-05-2021)
- [\*\*\*-21] <https://learn.sparkfun.com/tutorials/i2c/all>  
(Consultat la data: 04-05-2021)
- [POP-11] Popescu D., Ștefănoiu D. – Automatica Industrială, Editura AGIR, București, România, 2011.
- [DUM-10] Dumitrache Ioan - Ingineria Reglării Automate, Editura Politehnica Press, București, România, 2010.